Classification in Twitter via Compressive Sensing

Dimitrios Milioris

Classification in Twitter via Compressive Sensing

Dimitris Milioris
Bell Labs, Alcatel-Lucent & École Polytechnique Paris
Email: dimitrios.milioris@alcatel-lucent.com

***Abstract*—In this paper we introduce a novel low dimensional method to perform topic detection and classification in Twitter. The proposed method first employs Joint Complexity to perform topic detection. Then, based on the nature of the data, we apply the theory of Compressive Sensing to perform topic classification by recovering an indicator vector, while reducing significantly the amount of information from tweets. In this paper we exploit datasets in various languages collected by using the Twitter streaming API, and achieve increased classification accuracy when comparing to state-of-the-art methods based on bag-of-words, along with several reconstruction techniques.**

I. Introduction

During the last decade, the information within social networks has increased dramatically. The ability to study the interaction and communication between users in these networks can provide real time valuable prediction of the evolution of the information.

In this paper, first we use the theory of Joint Complexity (JC) to perform topic detection. The evaluation of the proposed method is based on the detection of real world topics like the categories of a mainstream news portal. We use large datasets, which are tweets from politics, economics, sport, technology and lifestyle. We decompose the tweets in linear time into a memory efficient structure called Suffix Tree and by overlapping two trees, in linear or sublinear average time, we obtain the JC defined as the cardinality of factors (subsequences) that are common in both trees. Then we classify new tweets into these categories with the power of Compressive Sensing (CS) by taking advantage of the nature of the data. The method is simple, context-free, with no grammar and no language assumptions, and does not use semantics. Therefore, there is no need of any specific dictionary or stemming process which will explode the complexity.

The paper is organised as follows: Section II introduces the JC method. Section III describes the application for classification via CS, while Section IV evaluates the performance with real data obtained from Twitter.

II. Joint Complexity

In a prior work, we extend JC estimate to Markov sources of any order on a finite alphabet. Markov models are more realistic and have a better approximation for text generation than memoryless sources [1], [2]. We derived a second order asymptotics for JC of the following form

$$\gamma \frac{n^{\kappa}}{\sqrt{\alpha \log n + \beta}}, \tag{1}$$

for $\kappa < 1,\ \gamma,\ \alpha > 0,\ \beta > 0$, with n being the length of the sequence. This estimate has a faster convergence, and is preferred for texts of order $n \approx 10^2$; Therefore JC is an efficient method to capture the similarity degree of short texts, *e.g.* tweets (< 140 characters).

The topic detection method based on JC proceeds in two steps. First, we construct the training databases (DBs) by using Twitter's streaming API (using the basic *.json* format) while filtering for specific keywords. Using these requests we build C classes on different topics. Assume that each class contains N tweets (*eg.* $C = 5$, i.e. Classes: politics, economics, sports, technology, lifestyle of $N = 12,000$ tweets). We populate each class by allocating a number of keywords. Assume that we have a dataset of S timeslots with $s = 1 \dots S$, and each timeslot is a 15 minutes request in Twitter API. For every tweet x_i, where $i = 1 \dots N$, with N being the total number of tweets, in the s-th timeslot, *i.e* x_i^s, we build a Suffix Tree.

Then we compute the JC metric, $JC(x_i^s, x_j^s)$ of the tweet x_i^s with every other tweet x_j^s of the s-th timeslot, where $j = 1 \dots N$, and $j \neq i$ (by convention we choose $JC(x_i^s, x_i^s) = 0$). For the S timeslots we store the JC scores in the matrices $S_1, S_2, \dots, S_S$ of $N \times N$ dimensions. We then assign a new tweet to the class that maximises the JC metric in that class. In order to limit the size of each reference class we delete the oldest tweets or the least significant ones (e.g. the ones which obtained the lowest JC score). This ensures the *low cost* and *efficiency* of the method.

III. Compressive Sensing

Let us first describe the main theoretical concepts of CS [3] as applied in the context of classification. Let $\mathbf{x} \in \mathbb{R}^N$ denote the signal of interest. Such signal can be represented as a linear combination of a set of basis $\{\psi_i\}_{i=1}^N$. By constructing a $N \times N$ basis matrix $\mathbf{\Psi} = [\psi_1, \psi_2, \dots, \psi_N]$, the signal x can be expressed as $x = \sum_{i=1}^N s_i \psi_i = \mathbf{\Psi} s$. In fact the signal is represented as $x = \mathbf{\Psi} s + \theta$, with $\theta \in \mathbb{R}^N$ being the noise, where $\mathbb{E}(\theta) = 0$ and $var(\theta) = O(|\Psi s|)$.

The efficiency of a CS method for signal approximation or reconstruction depends highly on the sparsity structure of the signal in a suitable transform domain associated with an appropriate sparsifying basis $\mathbf{\Psi}$. The measurement model in the original space-domain is expressed as $\mathbf{g} = \mathbf{\Phi x}$, where $\mathbf{g} \in \mathbb{R}^M$ is the measurement vector and $\mathbf{\Phi} \in \mathbb{R}^{M \times N}$ denotes the measurement matrix. The measurement model has the following equivalent transform-domain representation

$$\mathbf{g} = \mathbf{\Phi \Psi s} + \mathbf{\Phi \theta}\ . \tag{2}$$

In fact when the length of the sequence $n \to \infty$ and $N \to \infty$, $\mathbb{E}(\mathbf{\Psi} s) = O(nN)$, with $var(\theta) = O(nN)$,

$std(\theta) = O(\sqrt{|\Phi|n})$ and $\mathbb{E}(\mathbf{\Phi\theta}) = 0$. The second part of (2), $\mathbf{\Phi\theta}$ is of relative order $O(\frac{1}{\sqrt{nN}})$, and is negligible compare to $\mathbf{\Phi\Psi s}$ due to the law of large numbers.

In the framework of CS, the problem of classifying a tweet is reduced to a problem of recovering the one-sparse vector $\mathbf{s}$. Of course in practice we do not expect an exact sparsity, thus, the estimated class corresponds simply to the largest-amplitude component of $\mathbf{s}$. According to [4], [5], $\mathbf{s}$ can be recovered perfectly with high probability by solving the following optimization problem

$$\hat{\mathbf{s}} = \arg\min_{\mathbf{s}} \Big(\|\mathbf{s}\|_1 + \tau \|\mathbf{g} - (\mathbf{\Phi\Psi s})\|_2 \Big) \ , \qquad (3)$$

where τ is a regularization factor that controls the trade-off between the achieved sparsity and the reconstruction error.

During the training phase, we built our classes as described in Section II and for each class we extract the most representative tweet(s) (CTs) based on the Joint Complexity method. The vector $\mathbf{\Psi}_T^i$ consists of the highest JC scores of the i-th CT. The matrix $\mathbf{\Psi}_T$ is used as the appropriate sparsifying dictionary for the training phase. Moreover, a measurement matrix $\mathbf{\Phi}_T^i$ is associated with each transform matrix $\mathbf{\Psi}_T^i$.

A similar process is followed during the runtime phase. More specifically, we denote $\mathbf{x}_{c,R}$ as the Joint Complexity score of the incoming tweet with the CT_i classified at the current class c, where R denotes the runtime phase. The runtime CS measurement model is written as

$$\mathbf{g}_c = \mathbf{\Phi}_R \mathbf{x}_{c,R} \ , \qquad (4)$$

where $\mathbf{\Phi}_R$ denotes the corresponding measurement matrix during the runtime phase. The measurement vector $\mathbf{g}_c$ is formed for each CT_i according to (4) and the reconstruction takes place via the solution of (3), with the training matrix $\mathbf{\Psi}_T$ being used as the appropriate sparsifying dictionary.

IV. Experimental Results

The efficiency of the proposed classification method is evaluated on sets of tweets acquired from Twitter, while the classification accuracy of the tested methods was measured with the standard *F-score* metric, using a Ground Truth (GT) on more than $1,041,000$ tweets [6]. We selected the Document-Pivot (DP) technique to compare with our method, since it outperformed the other state-of-the-art techniques in a Twitter context as shown in [7]. More specifically, we used (a) Document Pivot (DP), (b) Joint Complexity with Compressive Sensing (JC+CS), (c) Document Pivot with URL (DPurl), (d) Joint Complexity and Compressive Sensing with URL (JCurl+CS), where (c) and (d) include the information of the compressed URL of a tweet concatenated with the original tweet's text and extracted from the *.json* file.

Fig. 1 compares the classification accuracy (increased by 27%) of the DP, DPurl and JC+CS, JCurl+CS method as a function of the percentage of measurements by using the ℓ_1-norm min. As we can see, JC with CS outperforms DP, and JCurl with CS outperforms DPurl. Fig. 2 compares the reconstruction performance between several widely-used norm-based techniques and Bayesian CS algorithms. More specifically, the following methods are employed[1]: 1) ℓ_1-norm min. (L1EQ-PD), 2) Orthogonal Matching Pursuit (OMP), 3) Stagewise Orthogonal Matching Pursuit (StOMP), 4) LASSO, 5) BCS, and 6) BCS-GSM [8], while BCS and BCS-GSM outperform the other introduced reconstruction techniques.

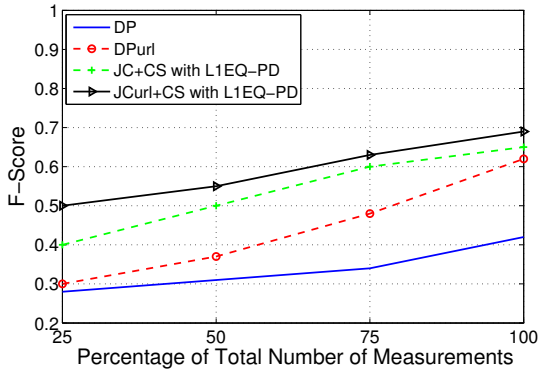

Fig. 1. Classification accuracy of various methods, measured by F-Score as a function of the percentage of measurements (%) by using the ℓ_1-norm min.

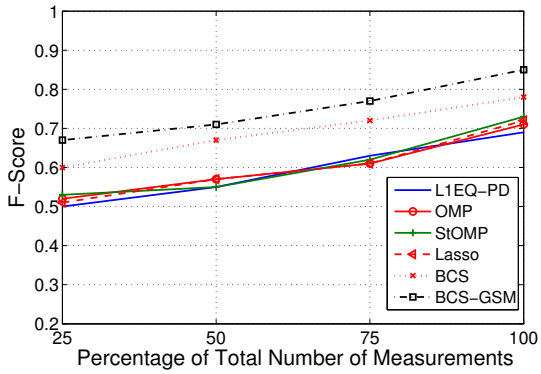

Fig. 2. Classification accuracy of the JCurl+CS method, measured by F-Score as a function of the percentage of measurements (%) by using several reconstruction techniques.

References

[1] P. Jacquet *et. al.*, "Classification of Markov Sources Through Joint String Complexity: Theory and Experiments", *in IEEE International Symposium on Information Theory (ISIT)*, Istanbul, Turkey, July 2013.

[2] D. Milioris and P. Jacquet, "Joint Sequence Complexity Analysis: Application to Social Networks Information Flow", *in Bell Labs Technical Journal*, Vol. 18, No. 4, 2014 (doi: 10.1002/bltj.21647).

[3] D. Milioris et al., "Low-dimensional signal-strength fingerprint-based positioning in wireless LANs", *in Ad Hoc Networks Journal, Elsevier*, Vol. 12, pp. 100–114, Jan. 2014.

[4] E. Candés, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *in IEEE Trans. on Information Theory*, Vol. 52, pp. 489–509, Feb. 2006.

[5] D. Donoho, "Compressive sensing", *in IEEE Trans. on Information Theory*, Vol. 52, No. 4, pp. 1289–1306, April 2006.

[6] S. Papadopoulos, D. Corney, and L. M. Aiello. "Snow 2014 data challenge: Assessing the performance of news topic detection methods in social media," in *WWW'14 Companion*, Seoul, Korea, Apr. 2014.

[7] L. M. Aiello *et al.*, "Sensing Trending Topics in Twitter", *in IEEE Transactions on Multimedia*, Vol. 15, Iss. 6, pp. 1268–1282, Oct 2013.

[8] G. Tzagkarakis, D. Milioris and P. Tsakalides, "Multiple-Measurement Bayesian Compressive Sensing using GSM Priors for DOA Estimation", in *35th Int. Conf. on Acoustics, Speech, and Sig. Proc. (ICASSP)*, Dallas, TX, Mar. 2010.

[1]For the implementation of methods 1)-5) the MATLAB codes can be found in: http://sparselab.stanford.edu/, http://www.acm.caltech.edu/l1magic, http://people.ee.duke.edu/~lcarin/BCS.html